GENETIC PROGRAMMING BASED HAAR-LIKE FEATURE CLASSIFIER DESIGN

Ondřej Benda
Master Degree Programme (5), FEEC BUT
E-mail: xbenda00@stud.feec.vutbr.cz

Supervised by: Jan Karásek
E-mail: karasek.jan@phd.feec.vutbr.cz

Abstract: This paper describes the use of the Genetic Programming algorithm for the Haar-like Feature classifier design. The classifier is applied for the Viola-Jones based image object detection instead of the cascade. The disadvantage of the cascade is the use of many features and one way pass only. The number of the features increases the detection time. The benefit of the decision tree classifier approach is the use of smaller amount of the features and therefore faster detection. The classifier is trained for the human face detection.

Keywords: Genetic Programming, Haar, Feature, Viola-Jones, Face, Detection

1 VIOLA-JONES FRAMEWORK

The Viola-Jones is the image object detection framework. It uses the integral image as image format representation, the haar-like feature as the object indicator (weak classifier) and the Adaboost as the learning algorithm. [1][2]

2 GENETIC PROGRAMMING BASED HAAR-LIKE FEATURE CLASSIFIER DESIGN

The traditional way of the Viola-Jones framework is to use the cascade. This work describes the use of Genetic Programming in order to evolve decision tree classifier as an object detector instead of the cascade.

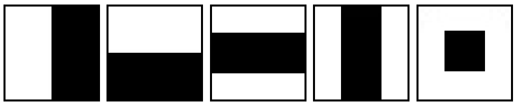
Genetic programming is the process of evolving programs on the principle of biological evolution where only the strongest ones survives.. The programs are represented as a trees composed of functions and terminals. Genetic Programming uses three genetic operators - crossover, mutation and reproduction.

In the algorithm, the tree is represented as a function (the root node) with two arguments (the descendant nodes) either the another function or the terminal. The terminals are the features, which can take only the boolean values (feature indicates the object or not). The functions are logic operations - AND, OR, NOR, NAND and XOR. The tree of three nodes and four leafs is described by equation 1.

$$f(img) = \mathbf{AND}(\mathbf{OR}(F_1(img), F_2(img)), \mathbf{XOR}(F_3(img), F_4(img)))) \tag{1}$$

The feature is a set of rectangles **R** defined by position coordinate, width, height and weight. The smaller rectangle compensates its size with the weight of the pixels - twice smaller rectangle has twice weighted pixels. The feature returns the difference of pixel sums under the rectangles weighted on one pixel of image in proportion of standard deviation of all image pixels.

$$F = \frac{\sum(R_1) * w_1 - \sum(R_2) * w_2}{\sigma * N} \tag{2}$$



Obrázek 1: Examples of features.

The features are trained on the set of the training images. These are gray-scale images of size 24x24 pixels. The training of feature means evolving such a threshold of the feature $t = F(img)$, which separates the positive and negative images most.

In the first step the features are generated. Then they are trained using the AdaBoost algorithm and the best ones are selected. The initial population is generated with the selected features by method called *Full* - each individual has full size.[4] The example of a tree in the generation is showed by equation 1.

All individuals are evaluated by the fitness function. The fitness function measures the probability of the right classification on the set of the training data (24x24 gray-scale images). The tree evaluates the image by processing the function which represents, on the given image. The variables values are the features indicating the object or not. The tree detects the object if the function based on the variables values returns the true.

The individuals are sorted by probability of right classification The individuals with higher probability has better chance to be selected for genetic operation.

The crossover selects two random subtrees in two selected trees and replace the first subtree by the second subtree. The mutation select a random subtree in selected tree and replace it with new random generated tree. The reproduction copy the selected tree directly into the new generation.

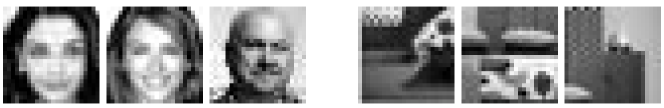
The crossover is applied with 0.7 probability, the mutation with 0.2 probability and the reproduction with 0.1 probability. The individuals from actual population are selected with the Roulette-Wheel selection method, which is fitness proportionate.

This process continues until the individual with desired probability of right classification is found or the time for the process is passed out. After the process of evolving ends, the final tree is evaluated on the testing set of the images.

3 RESULTS

The process of learning consists of two phases. First, the learning of the features with the AdaBoost algorithm. This phase takes about 12 hours of training time. The second phase is the training the strong classifier with the Genetic Programming. This part takes about 30 minutes.

The algorithm was trained for the human face detection on 4000 images - 2000 positive [6], 2000 negative [7]. The population has contained 300 individuals. Five-times more individuals has been created by the genetic operators in each generation and 300 selected to create new population. The stop conditions was: 0.96 probability of right decision or the process time 30 minutes. The final best individual was tested on the 2000 images - 1000 positive [6], 1000 negative [7].



Obrázek 2: Example of positive images (right) and negative images (left).

The best feature it self has classified with probability of right decision 0.72. The best individual in initial population has classified with probability of right decision 0.86. And the best individual in the last population has classified with probability of right decision 0.9412 - measured with the testing data.

Data type	Number of Features	AdaBoost	Genetic Programming	Precision
Faces	18 features	>10 hours	12 minutes	0.9412

Tabulka 1: Results Overview

4 CONCLUSION

The genetic programming has been tested to evolve classifier for the Viola-Jones object detection.

The evolved classifier reached 94.12 % accuracy of right decision on the set of the 2000 face images of size 24x24. The time of training classifier using the genetic programming is 12 minutes. The features training time using the AdaBoost algorithm is quite time demanding.

The solution to this may be to use the untrained features - not use the AdaBoost. Instead of the AdaBoost use the genetic programming and evolve the features along with the classifier. This is also idea for the future work.

REFERENCE

[1] Viola, P.; Jones, M.: *Robust Real-time Object Detection*, Vancouver, Canda, 2001.

[2] Jensen O. H.: *Implementing the Viola-Jones Face Detection Algorithm*, Kongens Lyngby, Denmark, 2008, ISBN 87-643-0008-0.

[3] Walker, M.: *Introduction to Genetic Programming*, 2001.

[4] Koza, J. R.: *Genetic Programming: On the Programming of Computers*

[5] POLI, Riccardo; LANGDON B. William; MCPHEE F. Nicholas; KOZA R. John; *A Field Guide to Genetic Programming* Lulu Enterprises, UK Ltd, 26.3.2008. 252 s. ISBN 978-1409200734. by Means of Natural Selection., 1992, The MIT Press.

[6] vis-www.cs.umass.edu; *Positive images dataset* [online] Available on URL: <http://vis-www.cs.umass.edu/lfw/#download>.

[7] cbcl.mit.edu; *Nagative images dataset* [online] Available on URL: <http://cbcl.mit.edu/cbcl/software-datasets/FaceData2.html>.